





0001 SUBROUTINE TIMRB (LUN)! NOTE BEGINNING OF TIMED INTERVAL  
0002  
0003 C Version: 'V04-000'  
0004 C  
0005 C\*\*\*\*\*  
0006 C\*  
0007 C\* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0008 C\* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0009 C\* ALL RIGHTS RESERVED.  
0010 C\*  
0011 C\* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0012 C\* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0013 C\* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0014 C\* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0015 C\* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0016 C\* TRANSFERRED.  
0017 C\*  
0018 C\* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0019 C\* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0020 C\* CORPORATION.  
0021 C\*  
0022 C\* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0023 C\* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0024 C\*  
0025 C\*  
0026 C\*\*\*\*\*  
0027 C  
0028 C++  
0029 C FACILITY: ERF, Errorlog Report Formatter  
0030 C  
0031 C ABSTRACT:  
0032 C  
0033 C Runtime statistics timing package.  
0034 C  
0035 C ENVIRONMENT:  
0036 C  
0037 C VAX/VMS operating system, unprivileged, user mode.  
0038 C  
0039 C MODIFIED BY:  
0040 C  
0041 C V03-002 SAR0212 Sharon A. Reynolds 22-Mar-1984  
0042 C Changed the carriage control in a format statement for  
0043 C use with output file.  
0044 C  
0045 C V03-001 JMG0010 Joel M. Gringorten 02-Feb-1984  
0046 C Rewrote the error handling from GETJPI system service to  
0047 C utilize LIB\$SIGNAL, and the ERFMSG file.  
0048 C  
0049 C  
0050 C SAVE CURRENT PROCESS STATISTICS IN VARIABLES IN COMMON  
0051 C USAGE:  
0052 C CALL TIMRB !START OF TIMED INTERVAL  
0053 C  
0054 C EQUATED SYMBOLS:  
0055 C  
0056 C  
0057 C COMMON /STAT\_VARS/ TO,BUFIO,CPUTIME,DIRIO,PFLTS

```
0058      INTEGER*4 BUFI0,CPUTIME,DIRIO,PFLTS
0059
0060      COMMON /JOB_PARAM/ LEN4A,BUFI0_CODE,BUFI0_ADR,ZERO,
0061                      2 LEN4B,CPUTIME_CODE,CPUTIME_ADR,ZERO1,
0062                      2 LEN4C,DIRIO_CODE,DIRIO_ADR,ZERO2,
0063                      2 LEN4D,PFLTS_CODE,PFLTS_ADR,ZERO3,
0064                      2 ZERO4
0065      BYTE      LUN
0066      INTEGER*2 LEN4A,LEN4B,LEN4C,LEN4D
0067      INTEGER*2 BUFI0_CODE,CPUTIME_CODE,DIRIO_CODE,PFLTS_CODE
0068      INTEGER*4 BUFI0_ADR,CPUTIME_ADR,DIRIO_ADR,PFLTS_ADR
0069      INTEGER*4 NEW_BUFI0,NEW_CPUTIME,NEW_DIRIO,NEW_PFLTS
0070      INTEGER*4 ZERO,ZERO1,ZERO2,ZERO3,ZERO4,SYSSGETJPI,STATUS
0071
0072      LOGICAL*1      ERROR
0073
0074      EXTERNAL      ERF_NOSTATS, ERF_GETJPIERR
0075
0076      C
0077      C      **** NOTE THE FOLLOWING CODES ARE VMS SYMBOLIC PARAMS.
0078      C      THEY MAY CHANGE IN FUTURE VERSIONS OF VMS...BEWARE!
0079      DATA BUFI0_CODE /1036/          JPI$_BUFI0
0080      DATA CPUTIME_CODE /1031/        JPI$_CPUTIM
0081      DATA DIRIO_CODE /1035/         JPI$_DIRIO
0082      DATA PFLTS_CODE /1034/         JPI$_PAGEFLTS
0083      DATA LEN4A,LEN4B,LEN4C,LEN4D /4,4,4,4/
0084
0085      C
0086      =====
0087      T0 = SECNDS(0.)
0088      BUFI0_ADR      = %LOC(BUFI0)
0089      CPUTIME_ADR    = %LOC(CPUTIME)
0090      DIRIO_ADR      = %LOC(DIRIO)
0091      PFLTS_ADR      = %LOC(PFLTS)
0092
0093      ERROR = .FALSE.
0094
0095      STATUS = SYSSGETJPI(.,,LEN4A,.,)
0096      IF (.NOT. STATUS) THEN
0097          CALL LIB$SIGNAL(ERF_GETJPIERR,%VAL(0),%VAL(STATUS))
0098
0099      ERROR = .TRUE.
0100      ENDIF
0101
0102      RETURN
0103
0104      ENTRY TIMRE (LUN)! PRINT EXECUTION STATISTICS FOR INTERVAL
0105
0106      C
0107      C      USAGE:      CALL TIMRE      !END OF TIMED INTERVAL
0108
0109      C      TIMRE OBTAINS PROCESS STATISTICS AND SUBTRACTS THE
0110      C      BEGINNING-OF-INTERVAL STATISTICS RECORDED BY TIMRB.
0111      C      THE INCREMENTAL VALUES ARE WRITTEN TO UNIT "TTY"
0112      C      (FORTRAN UNIT 6).
0113      C      BUFI0_ADR      = %LOC(NEW_BUFI0)
0114      C      CPUTIME_ADR    = %LOC(NEW_CPUTIME)
```

```
0115 DIRIO_ADR      = %LOC(NEW_DIRIO)
0116 PFLTS_ADR      = %LOC(NEW_PFLTS)
0117
0118 C
0119
0120 STATUS = SYSS$GETJPI(,,,LEN4A,,)
0121 IF (.NOT. STATUS) THEN
0122 CALL LIB$SIGNAL(ERF_GETJPIERR,%VAL(0),%VAL(STATUS))
0123
0124 ERROR = .TRUE.
0125 ENDIF
0126
0127 C
0128 CLKTIME = SECNDS(T0)
0129
0130 CPUSECS = (NEW_CPUTIME-CPUTIME)/100.
0131 BUFI0 = NEW_BUFI0 - BUFI0
0132 DIRIO = NEW_DIRIO - DIRIO
0133 PFLTS = NEW_PFLTS - PFLTS
0134
0135 IF (ERROR) THEN
0136 CALL LIB$SIGNAL(ERF_NOSTATS)
0137
0138 ELSE
0139
0140 CALL LINCHK (LUN,7)
0141
0142 110  WRITE(LUN,110) CPUSECS,CLKTIME,PFLTS,DIRIO,BUFI0
0143 FORMAT('PROGRAM RUNTIME STATISTICS',//,
0144 1 : ',T8,'TIMES IN SECONDS',T29,' PAGE',T39,'DIRECT',T49,'BUFFERED',//,
0145 2 : ',T12,'CPU ELAPSED',T29,'FAULTS',T39,' I/O',T49,' I/O',//,
0146 3 : ',T6,F9.1,F9.1,T25,I10,T39,I6,T49,I8,/)
0147 ENDIF
0148
0149 C  SAVE STATE FOR NEXT TIME AROUND
0150 C  THE USER IS SUPPOSED TO CALL TIMRB AGAIN, BUT IN CASE HE DOESN'T ...
0151
0152 T0 = SECNDS(0.)
0153 CPUTIME = NEW_CPUTIME
0154 BUFI0 = NEW_BUFI0
0155 DIRIO = NEW_DIRIO
0156 PFLTS = NEW_PFLTS
0157 RETURN
0158 END
```

## PROGRAM SECTIONS

Name	Bytes	Attributes
0 \$CODE	357	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 \$PDATA	175	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 \$LOCAL	116	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 STAT_VARS	20	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 JOB_PARAM	52	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	720	

## ENTRY POINTS

Address	Type	Name	Address	Type	Name
0-00000000		TIMRB	0-0000005F		TIMRE

## VARIABLES

Address	Type	Name	Address	Type	Name	Address	Type	Name	Address	Type	Name
3-00000004	I*4	BUFI0	4-00000004	I*4	BUFI0_ADR	4-00000002	I*2	BUFI0_CODE	2-00000018	R*4	CLKTIME
2-0000001C	R*4	CPUSECS	3-00000008	I*4	CPUTIME	4-00000010	I*4	CPUTIME_ADR	4-0000000E	I*2	CPUTIME_CODE
3-0000000C	I*4	DIRIO	4-0000001C	I*4	DIRIO_ADR	4-0000001A	I*2	DIRIO_CODE	2-00000000	L*1	ERROR
4-00000000	I*2	LEN4A	4-0000000C	I*2	LEN4B	4-00000018	I*2	LEN4C	4-00000024	I*2	LEN4D
AP-00000004a	L*1	LUN	2-00000004	I*4	NEW_BUFI0	2-00000008	I*4	NEW_CPUTIME	2-0000000C	I*4	NEW_DIRIO
2-00000010	I*4	NEW_PFLTS	3-00000010	I*4	PFLTS	4-00000028	I*4	PFLTS_ADR	4-00000026	I*2	PFLTS_CODE
2-00000014	I*4	STATUS	3-00000000	R*4	TO	4-00000008	I*4	ZERO	4-00000014	I*4	ZERO1
4-00000020	I*4	ZERO2	4-0000002C	I*4	ZERO3	4-00000030	I*4	ZERO4			

## LABELS

Address	Label
1-00000008	110'

## FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name	Type	Name	Type	Name	Type	Name
	ERF_GETJPIERR		ERF_NOSTATS	R*4	FOR\$SECNDS		LIB\$SIGNAL		LINCHK	I*4	SYSS\$GETJPI

## COMMAND QUALIFIERS

FORTRAN /LIS=LIS\$:TIMRB/OBJ=OBJ\$:TIMRB MSRC\$:TIMRB

```
/CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
/DEBUG=(NOSYMBOLS,TRACEBACK)
/STANDARD=(NOSYNTAX,NOSOURCE FORM)
/SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
/F77 /NOG_FLOATING /I4 /OPTIMIZE /WARNINGS /NOD_LINES /NOCROSS_REFERENCE /NOMACHINE_CODE /CONTINUATIONS=19
```

TIMRB

K 7  
16-Sep-1984 00:15:42    VAX-11 FORTRAN V3.4-56  
5-Sep-1984 14:23:34    DISKSVMMASTER:[ERF.SRC]TIMRB.FOR;1 Page 5

COMPILATION STATISTICS

Run Time:            1.78 seconds  
Elapsed Time:        4.31 seconds  
Page Faults:        132  
Dynamic Memory:     169 pages

0154 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

SBI LIS	TIMCMP LIS	TRNS.BITS LIS	TUTAPE LIS	UBAERR LIS	UBA LIS	UDEFIN LIS	UNKN.DISP LIS
STSEVENT LIS	TIMRB LIS	TU81SENSE LIS	UBA LIS	UDEFIN LIS	UNKN.DISP LIS	UNKNOWN LIS	UNKNOWN LIS
SUMMARY LIS	SYSPWRFL LIS	SYSTARTUP LIS	UBAINT LIS	UNKNOWN LIS	UNKNOWN LIS	UNKNOWN LIS	UNKNOWN LIS
SHRVECTOR LIS	TOF LIS	TSTAPE LIS	UNKNOWN LIS	UNKNOWN LIS	UNKNOWN LIS	UNKNOWN LIS	UNKNOWN LIS